

CERTIFICATE OF MAILING BY "EXPRESS MAIL"

"Express Mail" Mailing Label Number
EF238909431US

Date of Deposit 11/08/2001

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" Service under 37 CFR §1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

Lynn E. Cabiles

(Typed or printed name of person mailing)

Lynn E. Cabiles

(Signature of person mailing)

METHODOLOGY FOR FAST FILE TRANSFER PROTOCOL

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] This invention relates to data transmission and more particularly relates to fast and reliable transmission of files from one computer system to another computer system.

2. Discussion of the Related Art

[0002] A common protocol for computer networks is TCP/IP, which is the protocol used on the Internet. TCP stands for Transmission Control Protocol, and IP stands for Internet Protocol. TCP is a transport protocol that is responsible for end-to-end delivery of information across an internetwork (i.e., a network of networks).

[0003] File Transfer Protocol (FTP) may run as an application on top of TCP. FTP may support point-to-point file transfers. FTP may also depend on TCP for reliable delivery as TCP is a connection-oriented acknowledged transport protocol.

[0004] Connection-oriented implies that TCP first establishes a connection between two systems that intend to exchange data. Since most networks are built on shared media (for example, several systems sharing the same cabling), it may be necessary to break data into manageable pieces so that no two communicating computers monopolize the network. These pieces may be called packets. When an application sends a message (file or data) to TCP for transmission, TCP may break the message into packets, sized appropriately for the network, and send the packets over the network.

[0005] Because a single message is often broken into many packets, TCP marks these packets with sequence numbers before sending them across the network. The sequence numbers allow the receiving system to properly reassemble the packets into the original message. Being able to reassemble the original message is not enough, as the accuracy of the data should be verified. TCP may accomplish this by computing a checksum. The checksum is a mathematical computation applied (by the sender) to the data contained in the TCP packet. The receiving system may perform the same calculation on the received data and compare the result with the checksum that the sending system computed. If the results match, then the receiving system may send an acknowledgment (ACK) back to the originating system (or sending system). If

the results do not match, then the receiving system may ask the sending system to resend the packet.

[0006] TCP may also use port IDs to specify which application running on the system is sending or receiving the data. The port ID, checksum, and sequence number may be inserted into the TCP packet in a section called the header. The header is at the beginning of the packet containing this and other "control" information for TCP.

[0007] One problem with file transfers involves the length of time to transfer a file. For example, if one desires to transfer a file (such as a 2 Gigabyte file) across a latent network of great distance (such as between North America and Australia), then upwards of 20 to 40 hours may be needed to complete the total transfer of the file. This is due in part because it may take approximate 500 milliseconds for transmission of a packet and the return receipt of the acknowledgement back to the originating system. It is therefore desirable to speed up the file transfer time.

SUMMARY OF THE INVENTION

[0008] A method is provided for transferring a file over a network. This may include dividing a file into a plurality of blocks, transferring a first one of the plurality of blocks from a first entity and across the network toward a second entity. A second one of the plurality of blocks may be transmitted from the first entity and across the network toward the second entity while the first one of the plurality of blocks is being transferred across the network to the second entity.

[0009] A method may also be provided for transferring a file from a first entity to a second entity over a network. This may include receiving a first one of a plurality of blocks at the second entity, receiving a second one of the plurality of blocks at the second entity, receiving a last one of the plurality of blocks at the second entity, and assembling the first one of the plurality of blocks and the second one of the plurality of blocks into a single file at the second entity after receiving the last one of the plurality of blocks.

[0010] Embodiments of the present invention may utilize Transmission Control Protocol (TCP) for transmission of the plurality of blocks.

[0011] Other salient features of the invention will become apparent from the following detailed description taken in conjunction with the annexed drawings that disclose preferred embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] The foregoing and a better understanding of the present invention will become apparent from the following detailed description of example embodiments and the claims when read in connection with the accompanying drawings, all forming a part of the disclosure of this invention. While the foregoing and following written and illustrated disclosure focuses on disclosing example embodiments of the invention, it should be clearly understood that the same is by way of illustration and example only and the invention is not limited thereto.

[0013] The following represents brief descriptions of the drawings in which like reference numerals represent like elements and wherein:

[0014] FIG. 1 illustrates two computer systems coupled together by a communications network;

[0015] FIG. 2 is a block diagram of a computer system in which at least a portion of the present invention may be embodied;

[0016] FIG. 3 illustrates various layers of the computer system used for fast file transfer protocol in accordance with an example embodiment of the present invention;

[0017] FIG. 4 is a flowchart illustrating a method for transmitting a file in accordance with an example embodiment of the present invention; and

[0018] FIG. 5 is a flowchart illustrating a method for receiving a file in accordance with an example embodiment of the present invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0019] In the following detailed description, like reference numerals and characters may be used to designate identical, corresponding, or similar components in differing drawing figures. Furthermore, in the detailed description to follow, example sizes/models/values/ranges may be given, although the present invention is not limited thereto. Well-known elements have not been shown within the drawing figures for simplicity of illustration and discussion and so as not to obscure the invention.

[0020] A method may be provided for transferring a file over a network. This may include dividing a file into a plurality of blocks, transferring a first one of the plurality of blocks from a first entity and across the network toward a second entity. The method may also include transferring a second one of the plurality of blocks from the first entity and across the network toward the second entity while the first one of the plurality of blocks is being transferred across the network to the second entity. This transmission may be done in accordance with a Transmission Control Protocol.

[0021] FIG. 1 illustrates a first computer system 10 and a second computer system 30 that may be coupled together by a communications network 20 such as a latent network. For example, the communications network 20 may be a computer network (e.g., a LAN, a WAN or the internet), a wireless network (e.g., a cellular data network), some combination of these two types of communication mediums or some other communication medium.

[0022] As one example that will be used in the following discussion, the first computer system 10 may be provided within North America and the second computer system 30 may be provided within Australia. Thus, the first computer system 10 and the second computer system 30 may be separated by a considerable distance. The communications network 20 may be a satellite link or a landline, for example.

[0023] FIG. 2 is a block diagram of a computer system (such as the first computer system 10 or the second computer system 30) in which a portion of the present invention may be embodied. Other embodiments and configurations of

the computer system are also within the scope of the present invention. More particularly, each of the computer systems 10 and 30 may be computers such as PCs or workstations running on any of a variety of operating systems. The computer system may include a processor unit 50, a main memory unit 52 for storing programs and/or data, an input/output controller 54, a network interface 56, one or more input devices 58 (such as a keyboard and the mouse), a display device 60, a fixed or hard disk drive unit 62, a floppy disk unit 64, a tape drive unit 66 and a data bus 68 coupling these components to allow communication therebetween. The configuration of the computer system is not limited to the configuration shown in FIG. 2.

[0024] One or more computer programs may define the operational capabilities of the computer system. The computer programs may be loaded into a computer system via the hard drive 62, the floppy drive 64 and/or the tape drive 66. Alternatively, the computer programs may reside in a permanent memory location (e.g., a ROM chip) of the main memory 52.

[0025] FIG. 3 illustrates various layers of the computer system in the software and the hardware in accordance with an example embodiment of the present invention. Other embodiments are also within the scope of the present invention. More specifically, FIG. 3 shows an application layer 200, a fast file transfer protocol (FFTP) layer 210, a file transfer protocol (FTP) layer 220, a transport protocol (TCP) layer 230, a device driver 240 and communications hardware 250. As is well known, the communications hardware 250 may be the physical means of sending data over lines (i.e., the electrical, mechanical and functional

control of data circuits). The transport protocol layer 230 and the file transfer protocol layer 220 may also correspond to well known layers in the art. The application layer 200 may contain functions for particular application services such as file transfer, remote file access and virtual terminals. Embodiments of the present invention may be provided within the fast file transfer protocol layer 210 although embodiments may also be provided within the application layer 200. The fast file transfer protocol layer 210 may be provided within the first computer system 10 and the second computer system 30 (and any other relevant computer system). The fast file transfer protocol layer 210 may be software containing program instructions to perform features of the present invention as will be described below.

[0026] As discussed above, one problem with FTP is that latent networks reduce the FTP transfer rates. That is, a file transfer over a latent network may involve the sending of packets from the originating system (such as the first computer system 10) and the transmission of an acknowledgment back to the originating system before subsequent packets can be sent from the originating system. This process may be slowed down due to the higher latency on the network thereby increasing the waiting time for acknowledgements.

[0027] Embodiments of the present invention may divide (or break) a large file (such as a data file or program file) into multiple pieces (or blocks) and transfer each of the blocks over the network, rather than executing a single transfer of a large file. Any of several well-known FTP applications may be used to transmit the blocks across the network. The blocks may be reassembled at the receiving

system (such as the computer system 30) after they have been transferred. The software may be written so as to control the activity of the transfer and the rebuilding of the file blocks (or pieces).

[0028] The files may correspond to any number of types of files including data files, program files and visual files. Embodiments of the present invention are applicable to full motion visual products including the audio and video content. U.S. Patent Application No. 09/536,110, filed March 24, 2000 (entitled "Integrated Digital Production Line For Full-Motion Visual Products"), the subject matter of which is incorporated herein by reference, discloses a process and apparatus for managing visual content over a network. Embodiments of the present invention may be incorporated within the disclosed arrangements of that application.

[0029] Fast FTP may be incorporated within software and may increase the relative bandwidth of a file transfer, which thereby decreases the time of the transfer. That is, fast FTP may take a file and break (or divide) it into multiple blocks (or pieces) where each block may be the same size, except the last block which may be slightly larger as will be described. Each block may be transferred one at a time with a set interval between each transfer. This interval allows the transfers to start at different times so as not to stress the system. The Internet Protocol (IP) address of the receiving system (such as the second computer system 30) may also be passed to the FFTP layer 210. The Internet Protocol (IP) address may inform the FFTP which system to send the file blocks. A small control file may also be transmitted to assist in assembly of the blocks at the receiving system. On the receiving system, a separate FFTP application may be

launched to reassemble the transferred file. Therefore, when all the blocks of the file have reached the receiving system, the receiving system may recombine the blocks back into the original file. Stated differently, the FFTP may transfer multiple blocks (or pieces) of a file at a single time so as to fill up the available bandwidth or network capacity. Once a file (such as a 2 Gigabyte file) is broken into smaller blocks, a first block may be transmitted to the receiving system. A second block may be transmitted to the receiving system after waiting a set interval from the transmission of the first block. The set interval may be to avoid overloading the processor. The first computer system 10 may transfer a third block after waiting a set interval from the transmission of the second block. This process may continue until all the blocks of the file have been transferred to the receiving system (such as the second computer system 30).

[0030] By utilizing embodiments of the present invention, the TCP layer 230 does not change its normal operational mode. That is, the TCP layer 230 still waits for acknowledgments from the second computer system 30 regarding the transmitted packets. However, embodiments of the present invention more effectively utilize the available bandwidth by starting block transmission at slightly different times and by “simultaneously” transmitting blocks across the network.

[0031] The last block of a file may be an intercontrol block that contains information regarding the transferred file such as the file size, the correct bit length and all the other pieces. Embodiments of the present invention are also applicable to the intercontrol block being other than the last block of the file. When all the pieces of a respective file are received at the second computer

system 30, then the second computer system 30 may reassemble the blocks and present the reassembled file to the user application. This may be accomplished by the routines within the software of the second computer system 30.

[0032] FIG. 4 is a flowchart illustrating a method for transmitting a file in accordance with an example embodiment of the present invention. Other orders of operation and embodiments are also within the scope of the present invention. The method shown in FIG. 4 may be performed by software within the computer system 10 (such as in the FFTP layer 210). More specifically, the method may start (at operation 300) and wait until a transmission request is received (operation 310). If a transmission request is received in operation 310, then the system may proceed to determine whether the transmission file is larger than a predetermined size in operation 320. If the transmission file is not larger than the predetermined size in operation 320, then the entire file may be transmitted in operation 330. On the other hand, if the transmission file is larger than the predetermined size in operation 320, then the file may be divided (or broken down) into a number of smaller files (or blocks) in operation 340. The smaller file may be transmitted in operation 350. The system may wait a predetermined period of time (in operation 360) before transmitting another small file. In operation 370, the system may determine if there are any other small files to be transmitted. If there are small files (or blocks) to be transmitted in operation 370, then another small file may be transmitted in operation 350. The operations 350, 360 and 370 may be repeated until all the small files have been transmitted. For example, a first small file may be initially transmitted from the first computer

system 10. The first computer system 10 may wait a predetermined amount of time before transmitting a second small file. The first computer system 10 may subsequently wait a further predetermined amount of time from the transmission of the second small file until beginning transmission of a third small file. This cycle may continue until the last block (such as the interconnect block) of the file is transmitted from the first computer system 10 to the second computer system 30. Once all the small files have been transmitted, the operation may return back to operation 310 to wait for a subsequent transmission request.

[0033] The above method was described with respect to a predetermined time period between each transmission. Embodiments of the present invention are also applicable to a different amount of waiting time between each of the respective transmissions. Additionally, the predetermined period of time may be measured from the end of the previous transmission or from the beginning of the previous transmission. Still further, the system may transmit each of the small files based on a predetermined time schedule from an initial starting point.

[0034] FIG. 5 is a flowchart illustrating a method for receiving data in accordance with an example embodiment of the present invention. Other orders of operation and embodiments are also within the scope of the present invention. The method shown in FIG. 5 may be performed by software within the second computer system 30 (such as in the FFTP layer 210). More specifically, the method may start (at operation 400) by receiving a transmission at the second computer system 30 in operation 410. The second computer system 30 may store a received block of data in operation 420. The second computer system 30

may make a determination of whether a last block is received in operation 430. If the last block is received in operation 430, then the second computer system 30 may assemble all the blocks into a single file in operation 440. The FFTP layer 210 may transmit the location of the file to the application program in operation 450. On the other hand, if the last block is not received in operation 430, then the method may return to the operation 410 in which a subsequent block transmission may be received.

[0035] Stated differently, the receiving system may receive the first block and determine if all the transmissions have been received. If the last block has not been received, then a second block may subsequently be received. Once all the blocks have been received at the receiving system, then the blocks may be assembled into a single file.

[0036] Accordingly, embodiments of the present invention may provide a fast file transfer protocol (FFTP) methodology that speeds up file transfers by "simultaneously" transferring portions of a file. Simultaneously may refer to the successive transmissions of the files (or blocks) across the network. Fast FTP may be scaleable because it uses standard file transfer protocols and can be run on a PC. Fast FTP may also be used within a virtual private network (VPN) to protect data being transferred.

[0037] Embodiments of the present invention may use more of the available bandwidth than previous file transferring methodologies. The methodology does not wait for sequential acknowledgments before transmitting subsequent portions of the file. That is, because the methodology operates in a multi-thread manner

(by sending multiple blocks of data), the methodology does not wait for each sequential acknowledgement even though the system does receive multiple acknowledgements that may or may not be in sequential order. The fast FTP methodology may move data at all times rather than waiting for sequential acknowledgments during the long length of time to transfer the entire file.

[0038] Any reference in the above description to “one embodiment”, “an embodiment”, “example embodiment”, etc., means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of such phrases in various places in the specification are not necessarily all referring to the same embodiment. Further, when a particular feature, structure, or characteristic is described in connection with any embodiment, it is submitted that it is within the knowledge of one skilled in the art to effect such feature, structure, or characteristic in connection with other ones of the embodiments. Furthermore, for ease of understanding, certain method procedures may have been delineated as separate procedures; however, these separately delineated procedures should not be construed as necessarily order dependent in their performance. That is, some procedures may be able to be performed in an alternative ordering, simultaneously, etc.

[0039] Further, embodiments of the present invention or portions of embodiments of the present invention may be practiced as a software invention, implemented in the form of a machine-readable medium having stored thereon at least one sequence of instructions that, when executed, causes a machine to

[0040] Using the embodiments of the present invention it is possible to reduce the transmission time required for large files without resorting to the use of faster and more expensive communications equipment. Therefore, utilizing standard communications software and hardware a user would realize a significant improvement in communications performance when transmitting and receiving large files using the aforementioned embodiments of the present invention.

[0042] Although the present invention has been described with reference to a number of illustrative embodiments thereof, it should be understood that

numerous other modifications and embodiments can be devised by those skilled in the art that will fall within the spirit and scope of the principles of this invention. More particularly, reasonable variations and modifications are possible in the component parts and/or arrangements of the subject combination arrangement within the scope of the foregoing disclosure, the drawings and the appended claims without departing from the spirit of the invention. In addition to variations and modifications in the component parts and/or arrangements, alternative uses will also be apparent to those skilled in the art.

TRW Docket No. 38-0012